Proofs as schedules

Emmanuel Beffara[1] Virgile Mogbil[2]

[1]IML, CNRS & Université d'Aix-Marseille

[2]LIPN, CNRS & Université Paris Nord

Chocola – 14 March 2013

EB and VM. *Proofs as executions.*
Proceedings of IFIP TCS, 2012.

Plan

Logic vs computation

- The *formulae as types* approach:

$$\begin{aligned} \text{formula} &\leftrightarrow \text{type} \\ \text{proof rules} &\leftrightarrow \text{primitive instructions} \\ \text{proof} &\leftrightarrow \text{program} \\ \text{normalization} &\leftrightarrow \text{evaluation} \end{aligned}$$

- The *proof search* approach:

$$\begin{aligned} \text{formula} &\leftrightarrow \text{program} \\ \text{proof rules} &\leftrightarrow \text{operational semantics} \\ \text{proof} &\leftrightarrow \text{successful run} \end{aligned}$$

Axiom and cut:

$$\frac{}{u \multimap v \vdash u : \downarrow A^{\perp}, v : \uparrow A} \qquad \frac{P \vdash \Gamma, \vec{x} : A \qquad Q \vdash \vec{x} : A^{\perp}, \Delta}{(\nu\vec{x})(P \mid Q) \vdash \Gamma, \Delta}$$

Multiplicatives:

$$\frac{P \vdash \Gamma, \vec{x} : A \qquad Q \vdash \vec{y} : B, \Delta}{P \mid Q \vdash \Gamma, \vec{x}\vec{y} : A \otimes B, \Delta} \qquad \frac{P \vdash \Gamma, \vec{x} : A, \vec{y} : B}{P \vdash \Gamma, \vec{x}\vec{y} : A \parr B}$$

Actions:

$$\frac{P \vdash \Gamma, \vec{x} : A}{u(\vec{x}).P \vdash \Gamma, u : \downarrow A} \qquad \frac{P \vdash \Gamma, \vec{x} : A}{\bar{u}(\vec{x}).P \vdash \Gamma, u : \uparrow A}$$

Typing the π-calculus in linear logic

Properties of the system

Good things:

- Typed processes cannot diverge or deadlock.
- Typing is preserved by reduction (up to structural congruence).
- Explains translations of the λ-calculus into the π-calculus.
- Extends to differential LL.

Good things:

- Typed processes cannot diverge or deadlock.
- Typing is preserved by reduction (up to structural congruence).
- Explains translations of the λ-calculus into the π-calculus.
- Extends to differential LL.

Shortcomings:

- Typed processes are confluent.
- Many well-behaved interaction patterns are not typable.

$$a.\bar{b} \mid b.\bar{c} \mid \bar{a}.c.d$$

A few observations

Proof normalization, aka *cut elimination*:

- the meaning of a proof is in its normal form,
- normalization is an *explicitation* procedure,
- it really wants to be confluent.

Interpretation of concurrent processes:

- the meaning is the *interaction*, the final (irreducible) state is less relevant,
- a given process may behave very differently depending on scheduling decisions.

Proofs as schedules

The principles of our new interpretation:

$$\begin{aligned}
\text{formula} &\leftrightarrow \text{type of interaction} \\
\text{proof rules} &\leftrightarrow \text{primitives for building schedules} \\
\text{proof} &\leftrightarrow \text{schedule for a program} \\
\text{normalization} &\leftrightarrow \text{evaluation}
\end{aligned}$$

What this is not:

- *Curry-Howard* for processes:
  proofs are not programs, but behaviours of programs
- *Proof search*:
  the dynamics is not in proof construction but in cut-elimination
- *Specification, verification*:
  only "may"-style properties can be expressed, currently

Non-determinism in concurrent processes

We consider a CCS-style process calculus.

$$
\begin{array}{rll}
P, Q := & 1 & \text{inaction} \\
& a.P & \text{perform } a \text{ then do } P \\
& P \mid Q & \text{interaction of } P \text{ and } Q \\
& (\nu a)P & \text{scope restriction}
\end{array}
$$

There is one source of non-determinism:
the pairing of associated events upon synchronization

$$
a.P \mid a.Q \mid \bar{a}.R \;\to\; \begin{cases} a.P \mid Q \mid R \\ P \mid a.Q \mid R \end{cases}
$$

Pairings

Definition

A *pairing* is an association between occurrences of dual actions

p_1 :

p_2 :

$$P = a.b.A \mid \bar{a}.c.B \mid \bar{b}.\bar{c}.C \mid a.\bar{c}$$

Definition

A *determinisation* of P along a pairing p is a renaming $\partial_p(P)$ of actions in P where names are equal only for related actions.

$$\partial_{p_1}(P) = a'.b'.\partial(A) \mid \bar{a}.c.\partial(B) \mid \bar{b}''.\bar{c}''.\partial(C) \mid a.\bar{c}$$
$$\partial_{p_2}(P) = a.b.\partial(A) \mid \bar{a}.c.\partial(B) \mid \bar{b}.\bar{c}.\partial(C) \mid a'.\bar{c}'$$

Pairings

Facts about pairings:

- each run induces a pairing
- runs are equivalent up to permutation of independent events iff they induce the same pairing
- if p is a consistent pairing of P then p is the unique maximal consistent pairing of $\partial_p(P)$

Hence pairings are *execution schedules* and determinized terms represent them inside the process language.

Logic will type these schedules.

Types of schedules:

$A, B :=$	$\langle a \rangle A$	do action a and then act as A
	$A \otimes B$	two independent parts, one as A, the other as B
	$A \parr B$	A and B are both exhibited, but correlated
	α	an unspecified behaviour
	$\alpha^{\perp}$	something that can interact with α

Transforming schedules:

$A_1, \ldots, A_n \vdash B$ behave as type B using one schedule of each type A_i

Axiom and cut:

$$\frac{}{1 \vdash \alpha^{\perp}, \alpha} \qquad \frac{P \vdash \Gamma, A \qquad Q \vdash A^{\perp}, \Delta}{P \mid Q \vdash \Gamma, \Delta}$$

Multiplicatives:

$$\frac{P \vdash \Gamma, A \qquad Q \vdash B, \Delta}{P \mid Q \vdash \Gamma, A \otimes B, \Delta} \qquad \frac{P \vdash \Gamma, A, B}{P \vdash \Gamma, A \parr B}$$

Actions:

$$\frac{P \vdash \Gamma, A}{a.P \vdash \Gamma, \langle a \rangle A}$$

The role of the axiom rule

Two-sided presentation

$$
\begin{prooftree}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$\bar{b} : \alpha \vdash \langle \bar{b} \rangle \alpha$}
\UnaryInfC{$a.\bar{b} : \alpha \vdash \langle a\bar{b} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$\bar{c} : \alpha \vdash \langle \bar{c} \rangle \alpha$}
\UnaryInfC{$b.\bar{c} : \langle \bar{b} \rangle \alpha \vdash \langle \bar{c} \rangle \alpha$}
\UnaryInfC{$b.\bar{c} \vdash \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \langle \bar{b} \rangle \alpha \vdash \langle \bar{b} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$d : \alpha \vdash \langle d \rangle \alpha$}
\UnaryInfC{$c.d : \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$c.d : \langle \bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\UnaryInfC{$\bar{a}.c.d : \langle a\bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$b.\bar{c} \mid \bar{a}.c.d : \langle a\bar{b} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$a.\bar{b} \mid b.\bar{c} \mid \bar{a}.c.d : \alpha \vdash \langle d \rangle \alpha$}
\end{prooftree}
$$

$$
\dfrac{
\dfrac{\dfrac{\dfrac{}{1 : \alpha \vdash \alpha}}{\bar{b} : \alpha \vdash \langle \bar{b} \rangle \alpha}}{a.\bar{b} : \alpha \vdash \langle a\bar{b} \rangle \alpha}
\qquad
\dfrac{
\dfrac{\dfrac{\dfrac{\dfrac{}{1 : \alpha \vdash \alpha}}{\bar{c} : \alpha \vdash \langle \bar{c} \rangle \alpha}}{b.\bar{c} : \langle \bar{b} \rangle \alpha \vdash \langle \bar{c} \rangle \alpha}}{b.\bar{c} \vdash \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha}
\qquad
\dfrac{\dfrac{\dfrac{}{1 : \langle \bar{b} \rangle \alpha \vdash \langle \bar{b} \rangle \alpha} \qquad \dfrac{\dfrac{\dfrac{}{1 : \alpha \vdash \alpha}}{d : \alpha \vdash \langle d \rangle \alpha}}{c.d : \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}}{c.d : \langle \bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}}{\bar{a}.c.d : \langle a\bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}
}{b.\bar{c} \mid \bar{a}.c.d : \langle a\bar{b} \rangle \alpha \vdash \langle d \rangle \alpha}
}{a.\bar{b} \mid b.\bar{c} \mid \bar{a}.c.d : \alpha \vdash \langle d \rangle \alpha}
$$

The role of the axiom rule

Two-sided presentation

$$
\dfrac{
\dfrac{\dfrac{\dfrac{}{1 : \alpha \vdash \alpha}}{\bar{b} : \alpha \vdash \langle \bar{b} \rangle \alpha}}{a.\bar{b} : \alpha \vdash \langle a\bar{b} \rangle \alpha}
\qquad
\dfrac{
\dfrac{\dfrac{\dfrac{\dfrac{}{1 : \alpha \vdash \alpha}}{\bar{c} : \alpha \vdash \langle \bar{c} \rangle \alpha}}{b.\bar{c} : \langle \bar{b} \rangle \alpha \vdash \langle \bar{c} \rangle \alpha}}{b.\bar{c} \vdash \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha}
\quad
\dfrac{\dfrac{\dfrac{}{1 : \langle \bar{b} \rangle \alpha \vdash \langle \bar{b} \rangle \alpha} \quad \dfrac{\dfrac{\dfrac{}{1 : \alpha \vdash \alpha}}{d : \alpha \vdash \langle d \rangle \alpha}}{c.d : \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}}{c.d : \langle \bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}}{\bar{a}.c.d : \langle a\bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}
}{b.\bar{c} \mid \bar{a}.c.d : \langle a\bar{b} \rangle \alpha \vdash \langle d \rangle \alpha}
}{a.\bar{b} \mid b.\bar{c} \mid \bar{a}.c.d : \alpha \vdash \langle d \rangle \alpha}
$$

$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{}{1 : \alpha \vdash \alpha}
}{\bar{b} : \alpha \vdash \langle \bar{b} \rangle \alpha}
}{a.\bar{b} : \alpha \vdash \langle a\bar{b} \rangle \alpha}
\qquad
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\dfrac{}{1 : \alpha \vdash \alpha}
}{\bar{c} : \alpha \vdash \langle \bar{c} \rangle \alpha}
}{b.\bar{c} : \langle \bar{b} \rangle \alpha \vdash \langle \bar{c} \rangle \alpha}
}{b.\bar{c} \vdash \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha}
\qquad
\dfrac{
\dfrac{
\dfrac{}{1 : \langle \bar{b} \rangle \alpha \vdash \langle \bar{b} \rangle \alpha}
\quad
\dfrac{
\dfrac{
\dfrac{}{1 : \alpha \vdash \alpha}
}{d : \alpha \vdash \langle d \rangle \alpha}
}{c.d : \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}
}{c.d : \langle \bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}
}{\bar{a}.c.d : \langle a\bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha}
}{b.\bar{c} \mid \bar{a}.c.d : \langle a\bar{b} \rangle \alpha \vdash \langle d \rangle \alpha}
}{a.\bar{b} \mid b.\bar{c} \mid \bar{a}.c.d : \alpha \vdash \langle d \rangle \alpha}
$$

The role of the axiom rule

Two-sided presentation

$$
\begin{prooftree}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$\bar{b} : \alpha \vdash \langle \bar{b} \rangle \alpha$}
\UnaryInfC{$a.\bar{b} : \alpha \vdash \langle a\bar{b} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$\bar{c} : \alpha \vdash \langle \bar{c} \rangle \alpha$}
\UnaryInfC{$b.\bar{c} : \langle \bar{b} \rangle \alpha \vdash \langle \bar{c} \rangle \alpha$}
\UnaryInfC{$b.\bar{c} \vdash \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \langle \bar{b} \rangle \alpha \vdash \langle \bar{b} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$d : \alpha \vdash \langle d \rangle \alpha$}
\UnaryInfC{$c.d : \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$c.d : \langle \bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\UnaryInfC{$\bar{a}.c.d : \langle a\bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$b.\bar{c} \mid \bar{a}.c.d : \langle a\bar{b} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$a.\bar{b} \mid b.\bar{c} \mid \bar{a}.c.d : \alpha \vdash \langle d \rangle \alpha$}
\end{prooftree}
$$

The role of the axiom rule

Two-sided presentation

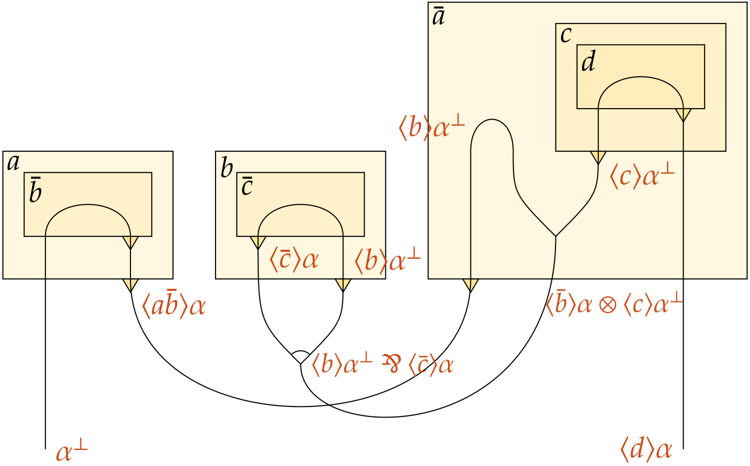
$$
\begin{prooftree}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$\bar{b} : \alpha \vdash \langle \bar{b} \rangle \alpha$}
\UnaryInfC{$a.\bar{b} : \alpha \vdash \langle a\bar{b} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$\bar{c} : \alpha \vdash \langle \bar{c} \rangle \alpha$}
\UnaryInfC{$b.\bar{c} : \langle \bar{b} \rangle \alpha \vdash \langle \bar{c} \rangle \alpha$}
\UnaryInfC{$b.\bar{c} \vdash \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \langle \bar{b} \rangle \alpha \vdash \langle \bar{b} \rangle \alpha$}
\AxiomC{}
\UnaryInfC{$1 : \alpha \vdash \alpha$}
\UnaryInfC{$d : \alpha \vdash \langle d \rangle \alpha$}
\UnaryInfC{$c.d : \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$c.d : \langle \bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\UnaryInfC{$\bar{a}.c.d : \langle a\bar{b} \rangle \alpha, \langle \bar{b} \rangle \alpha \multimap \langle \bar{c} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$b.\bar{c} \mid \bar{a}.c.d : \langle a\bar{b} \rangle \alpha \vdash \langle d \rangle \alpha$}
\BinaryInfC{$a.\bar{b} \mid b.\bar{c} \mid \bar{a}.c.d : \alpha \vdash \langle d \rangle \alpha$}
\end{prooftree}
$$

$$
\dfrac{
  \dfrac{
    \dfrac{
      \dfrac{}{1 \vdash \alpha^\perp, \alpha}
    }{\bar{b} \vdash \alpha^\perp, \langle \bar{b} \rangle \alpha}
  }{a.\bar{b} \vdash \alpha^\perp, \langle a\bar{b} \rangle \alpha}
  \qquad
  \dfrac{
    \dfrac{
      \dfrac{
        \dfrac{
          \dfrac{}{1 \vdash \alpha^\perp, \alpha}
        }{\bar{c} \vdash \alpha^\perp, \langle \bar{c} \rangle \alpha}
      }{b.\bar{c} \vdash \langle b \rangle \alpha^\perp, \langle \bar{c} \rangle \alpha}
    }{b.\bar{c} \vdash \langle b \rangle \alpha^\perp \parr \langle \bar{c} \rangle \alpha}
    \qquad
    \dfrac{
      \dfrac{
        \dfrac{}{1 \vdash \langle b \rangle \alpha^\perp, \langle \bar{b} \rangle \alpha}
        \qquad
        \dfrac{
          \dfrac{
            \dfrac{}{1 \vdash \alpha^\perp, \alpha}
          }{d \vdash \alpha^\perp, \langle d \rangle \alpha}
        }{c.d \vdash \langle c \rangle \alpha^\perp, \langle d \rangle \alpha}
      }{c.d \vdash \langle b \rangle \alpha^\perp, \langle \bar{b} \rangle \alpha \otimes \langle c \rangle \alpha^\perp, \langle d \rangle \alpha}
    }{\bar{a}.c.d \vdash \langle \bar{a} b \rangle \alpha^\perp, \langle \bar{b} \rangle \alpha \otimes \langle c \rangle \alpha^\perp, \langle d \rangle \alpha}
  }{b.\bar{c} \mid \bar{a}.c.d \vdash \langle \bar{a} b \rangle \alpha^\perp, \langle d \rangle \alpha}
}{a.\bar{b} \mid b.\bar{c} \mid \bar{a}.c.d \vdash \alpha^\perp, \langle d \rangle \alpha}
$$

Duality: $(A \otimes B)^\perp = A^\perp \parr B^\perp$, $(\langle a \rangle A)^\perp = \langle \bar{a} \rangle (A^\perp)$.

The role of the axiom rule

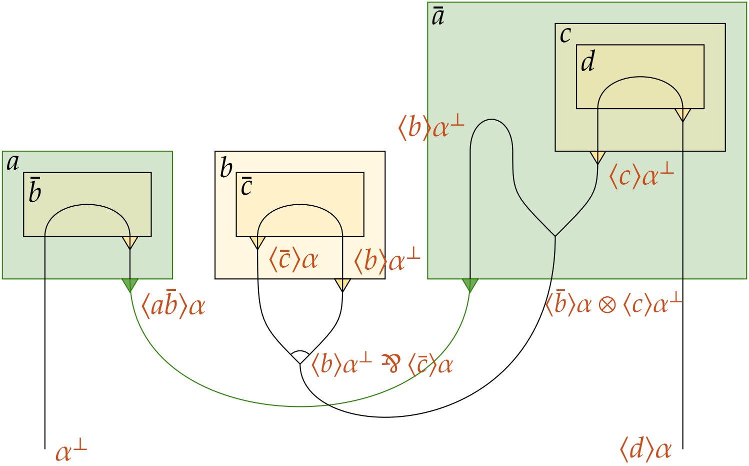
Proof net presentation



$$a.\bar{b}.1 \mid (b.\bar{c}.1 \mid \bar{a}.c.d.1)$$

The role of the axiom rule

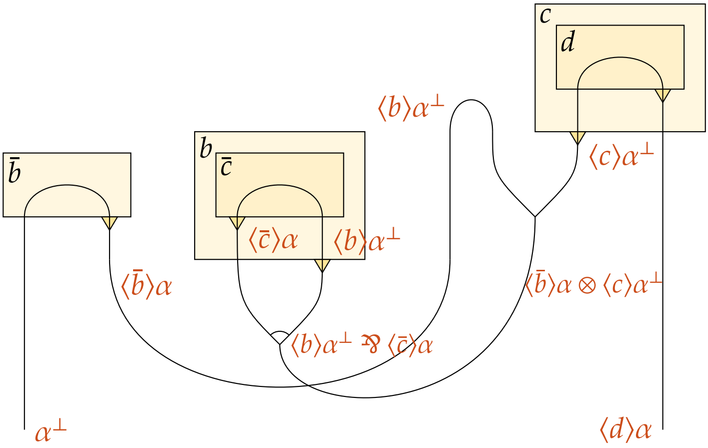
Proof net presentation



$$a.\bar{b}.1 \mid (b.\bar{c}.1 \mid \bar{a}.c.d.1)$$

The role of the axiom rule

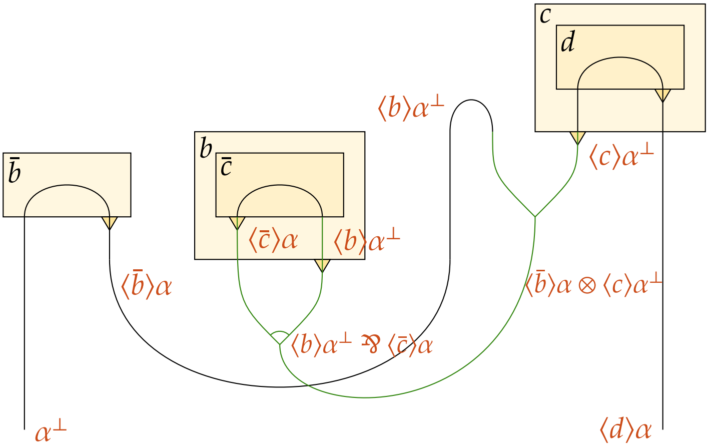
Proof net presentation



$$\bar{b}.1 \mid (b.\bar{c}.1 \mid c.d.1)$$

The role of the axiom rule

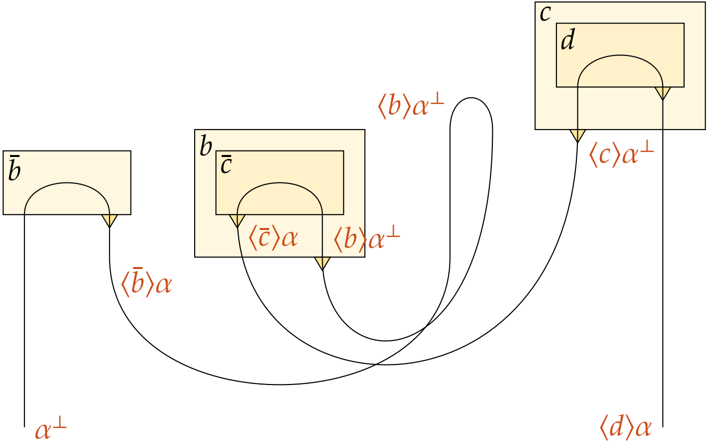
Proof net presentation



$$\bar{b}.1 \mid (b.\bar{c}.1 \mid c.d.1)$$

The role of the axiom rule

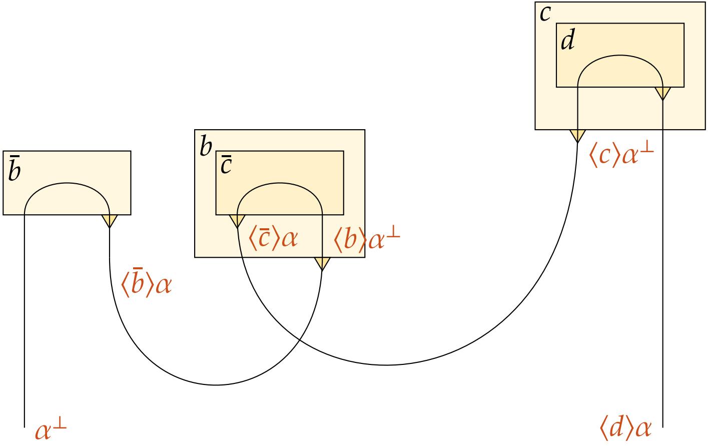
Proof net presentation



$$\bar{b}.1 \mid (b.\bar{c}.1 \mid c.d.1)$$

The role of the axiom rule

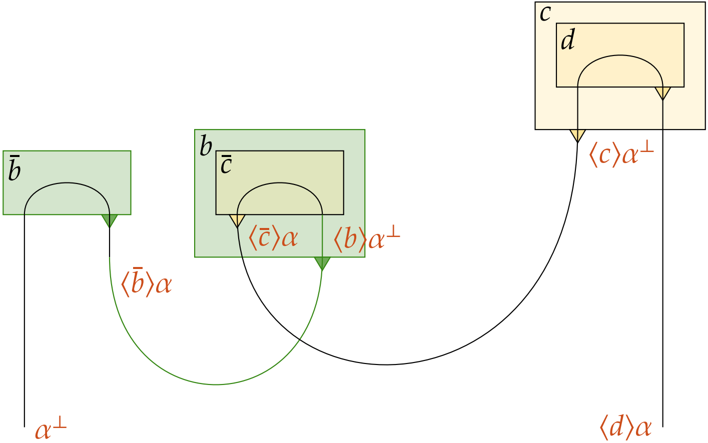
Proof net presentation



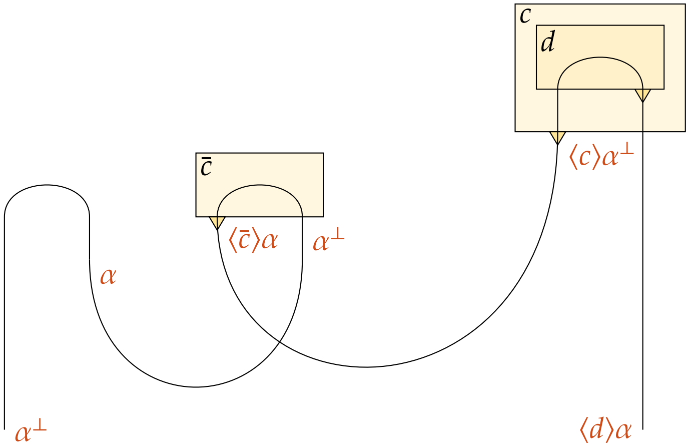
$$\bar{b}.1 \mid (b.\bar{c}.1 \mid c.d.1)$$

The role of the axiom rule

Proof net presentation



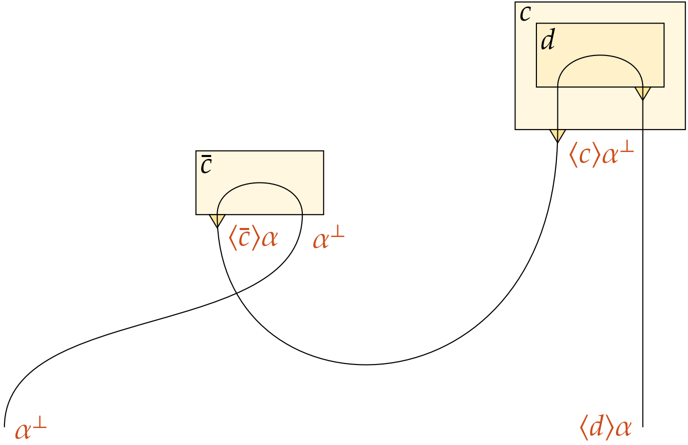
$$\bar{b}.1 \mid (b.\bar{c}.1 \mid c.d.1)$$



$$1 \mid (\bar{c}.1 \mid c.d.1)$$

The role of the axiom rule

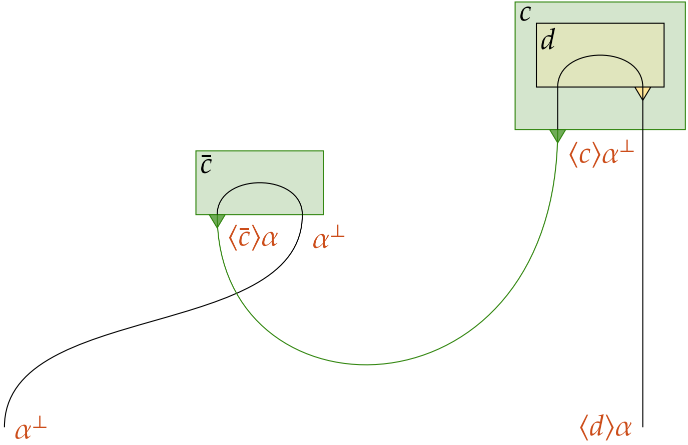
Proof net presentation



$$\bar{c}.1 \mid c.d.1$$

The role of the axiom rule

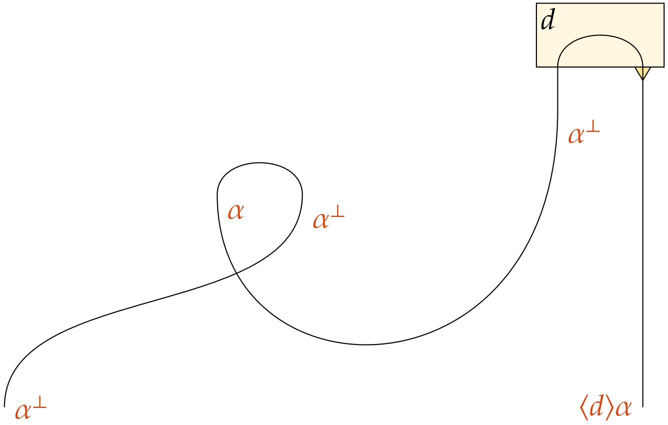
Proof net presentation



$\bar{c}.1 \mid c.d.1$

The role of the axiom rule

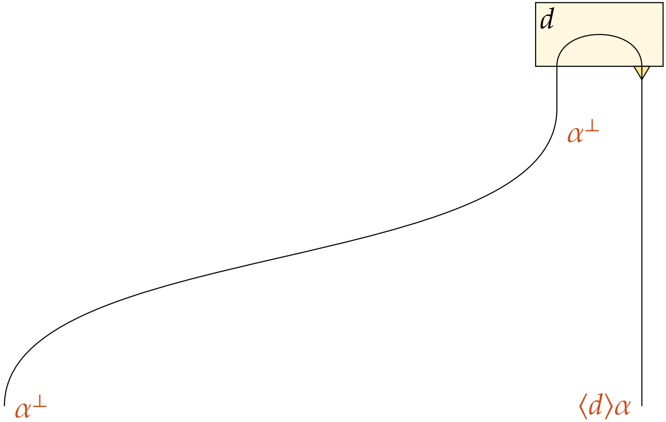
Proof net presentation



$1 \mid d.1$

The role of the axiom rule

Proof net presentation



$d.1$

Mandatory theorems

Theorem (Soundness)

Typing is preserved by reduction,
head cut-elimination steps correspond to execution steps.

- a typed deterministic term cannot deadlock,
- normalization corresponds to a particular execution.

Mandatory theorems

Theorem (Soundness)

Typing is preserved by reduction,
head cut-elimination steps correspond to execution steps.

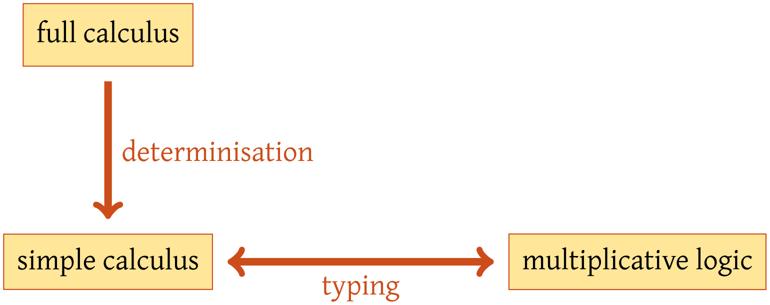
- a typed deterministic term cannot deadlock,
- normalization corresponds to a particular execution.

Theorem (Completeness)

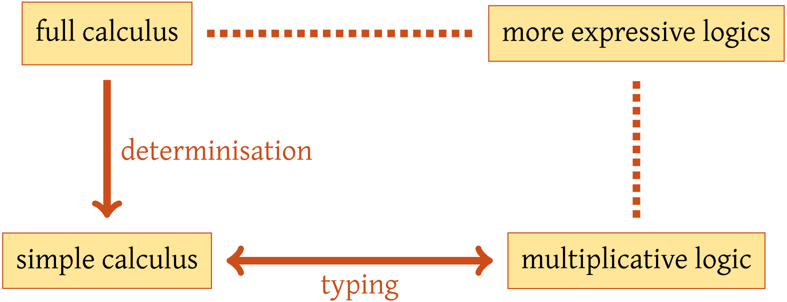
For every lock-avoiding run $P_1 \to \ldots \to P_n$ *there are corresponding typings such that* $\pi_1 : P_1 \vdash \Gamma \to \ldots \to \pi_n : P_n \vdash \Gamma$ *is a cut elimination sequence.*

- need to define "lock-avoiding"

Summing up

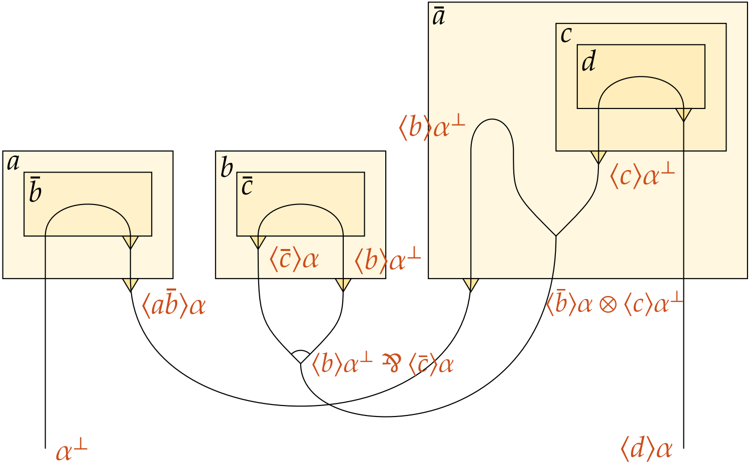

full calculus
determinisation
simple calculus
typing
multiplicative logic

Summing up


full calculus
more expressive logics
determinisation
simple calculus
typing
multiplicative logic

From one typing to the other

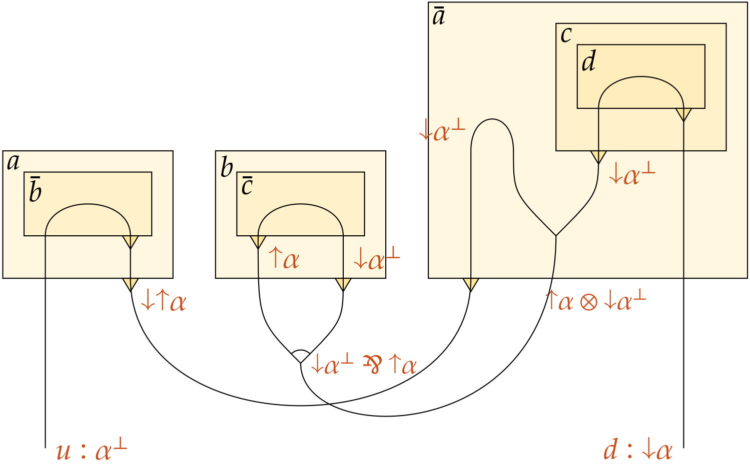
Continuation passing style in concurrency?



$$a \quad .\bar{b} \quad .1 \mid \quad \big(b \quad .\bar{c} \quad .1 \mid \bar{a} \quad .c \quad .d \quad .1\big)$$

From one typing to the other

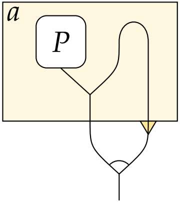
Continuation passing style in concurrency?



$$(\nu a)\Big(a(b).\bar{b}\langle u\rangle.1 \mid (\nu c)\big(b(y).\bar{c}\langle y\rangle.1 \mid \bar{a}\langle b\rangle.c(z).d\langle z\rangle.1\big)\Big)$$

Getting out of the box

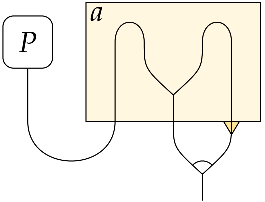
- It *seems* that all actions can be typed the same way:



$$\langle a\cdot\rangle A := \forall\alpha((A \multimap \alpha) \multimap \langle a\rangle\alpha) = \forall\alpha((A \otimes \alpha^{\perp}) \parr \langle a\rangle\alpha)$$

Getting out of the box

- It *seems* that all actions can be typed the same way:

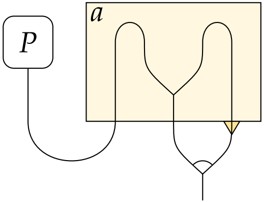


$$\langle a\cdot\rangle A := \forall\alpha((A \multimap \alpha) \multimap \langle a\rangle\alpha) = \forall\alpha((A \otimes \alpha^\perp) \parr \langle a\rangle\alpha)$$

- Cut expansion is acceptable at the logical level.

Getting out of the box

- It *seems* that all actions can be typed the same way:

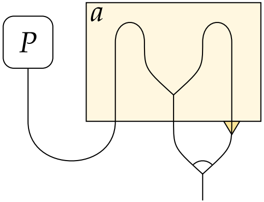


$$\langle a\cdot\rangle A := \forall\alpha((A \multimap \alpha) \multimap \langle a\rangle\alpha) = \forall\alpha((A \otimes \alpha^{\perp}) \parr \langle a\rangle\alpha)$$

- Cut expansion is acceptable at the logical level.
- Moreover, there is no real need for boxes in the confluent world!

Getting out of the box

- It *seems* that all actions can be typed the same way:



$$\langle a\cdot\rangle A := \forall\alpha((A \multimap \alpha) \multimap \langle a\rangle\alpha) = \forall\alpha((A \otimes \alpha^{\perp}) \parr \langle a\rangle\alpha)$$

- Cut expansion is acceptable at the logical level.
- Moreover, there is no real need for boxes in the confluent world!

$\to$ a boxless calculus, reminiscent of translations of π into solos

Conclusion, extensions

Current state of affairs:

- A logical description of scheduling in processes
  - describes how schedules can be safely composed
  - normal forms as basic *open* schedules
- Explicitation of *control flow* through processes
- Hints for a new study of *causality* in processes

Possible extensions:

- Connectives to combine related behaviours:

$$t_1.(t_2 + f_2 \mid \bar{t}_0) + f_1.(t_2.\bar{t}_0 + f_1.\bar{f}_0) \vdash B[t_1, f_1] \otimes B[t_2, f_2] \multimap B[t_0, f_0]$$

$$\text{where } B[t, f] := \alpha \multimap \langle \bar{t} \rangle \alpha \oplus \langle \bar{f} \rangle \alpha$$

- Predicates to describe states
- Richer action modalities for richer communication

C'est tout pour aujourd'hui.